Toward an Architecture for Ad Hoc Grids

Kaizar Amin*† Gregor von Laszewski*‡, Armin R. Mikler†
*Mathematics and Computer Science Division, Argonne National Laboratory, U.S.A.
†Computer Science and Engineering Department, University of North Texas, U.S.A.
‡gregor@mcs.anl.gov

Abstract— **The advantages offered by existing Grid frameworks have resulted in a wide range of applications adopting the Grid approach. The first generation of production Grids have focused on the creation of large virtual organizations that share high end resources as part of a static resource pool. However as many collaborative interactions take places on a sporadic or ad hoc fashion outside of the virtual organization, such Grids become impractical. In this paper, we outline an extension to the Grid architecture that addresses this issue. We refer to this architecture a as *sporadic* or *ad hoc Grid*. We discuss use cases that justify our efforts toward a self-organizing ad hoc Grid architecture. We outline the functional principles of this architecture and propose our framework to implement them.**

I. Introduction

The expansion of the Grid community from the scientific domain to include the commercial sector can be compared to the initial proliferation of the Internet. However, unlike a single global Internet, there exist several overlapping Grid architectures supporting different requirements and scale as discussed in [1]. In this paper, we discuss how the Grid architecture is differentiated by requirements posed by various user communities while focusing on ad hoc use modalities.

If we restrict our view to focus on organizational boundaries one way to classify existing computational Grid architectures at the coarse level is based on *national production Grids*, *community production Grids* [1], *enterprise production Grids*, and *volunteer production Grids* [2].

As per this classification, national production Grids aggregate high-end computing, data, and network resources across a nation to provide a unified distributed computing infrastructure [3], [4]. Membership, collaboration, and access to national Grids are regulated by the membership in a virtual organization sponsored on a national scale and are available to applications and groups of national importance.

Community production Grids are structurally similar to national Grids. Rather than aggregating resources on a national level, however, they represent a pool of resources across multiple geographic (potentially international) and administrative domains to achieve a mutually beneficial scientific or commercial goal of interest to the community [5]–[7]. Membership in a community Grid is usually controlled by a specially appointed administrative authority and is available only to member and collaborating organizations. Special cases of community production Grids are enterprise and volunteer production Grids.

Enterprise production Grids are restricted to resources that are part of the organization constituting the enterprise. This may include low-end computational resources such as desktops and laptops within a single organization as part of a powerful distributed computing framework at no additional hardware cost [8]. Access to an enterprise Grid is available only to the members of the enterprise and is most often restricted to profit-making enterprise applications.

Volunteer production Grids allow Internet users to altruistically donate unused computational cycles to achieve, most often, a nonprofit scientific task [9]. In contrast to traditional community Grids, the membership is based on an implicit trust model that is established through an inverse security assurance. While in traditional Grids, the users run their applications on trusted resources; in a volunteer Grid the resource contributors execute trusted applications. Internet users can contribute resources to the volunteer Grid. However, consumption of these resources is restricted to the controlling organization or service employing a master-slave computing model.

Despite these differences in the *Gestalt of the Grid* [1]. based on scale and motivations by the Grid users, the underlying Grid architectures share some common traits. First, they support mutually collaborative communities. Irrespective of their organizational orientation, participants of these Grid architectures share a synchronized non-conflicting objective. Second, all of these architectures adopt a centralized and regulated control for membership and access privileges [10]. They have a dedicated administrative authority responsible for the policy enforcement, monitoring, and maintenance of Grid resources. Third, they assume a stable and well-defined collaboration. Grid collaborations are accompanied by agreed-upon policies regarding the usage, privileges, and application deployment on these Grids. Due to the organizational involvements and legal implications, considerable effort is put into formulating these policies, which rarely change during the lifetime of a Grid collaboration.

Nevertheless, several applications do require competing communities or communities that *continuously change* their usage policies, membership and goals during the lifetime of the Grid.

Although the ad hoc and sporadic nature of Grids were already observed within the very first documented Globus/Grid application [11], current Grid architectures still fail to support certain aspects of this class of collaborative applications. Motivated by the need to support such applications, we propose an enhancement to the commodity Grid architecture that is capable of handling sporadic and ad hoc communities and collaborations with dynamically changing membership and access policies. We refer to this architecture as *sporadic* or *ad hoc Grid* [12].

The rest of this paper describes the motivation, requirements, and functionality of ad hoc Grids in more detail. Section II describes additional applications that are not yet supported by existing Grid architectures, thereby motivating the need for ad hoc Grids. Section III provides a functionality overview of ad hoc Grids. Section IV introduces our proposed framework for addressing several issues relevant in developing a practical commodity ad hoc Grid architecture. Section V summarizes the motivation for and the advantages offered by ad hoc Grids.

II. Motivating Use Cases

Several applications and use cases can be identified in practice that cannot be accomplished with traditional Grid frameworks. In this section we discuss some of the use cases that motivate the idea and development of ad hoc Grids.

A. Transient Collaborations of Peers

Consider the following use case. A group of geographically separated scientists require ad hoc, short-term collaboration and resource sharing in a secure environment to evaluate different experimental simulations of an application [11]. Assume, one scientist contributes a propriety simulation service, one pools a unique visualization service to render the results of the simulated experiment, another scientist provides a data repository storing the input datasets for which he owns the intellectual property, and a few others want to interactively discuss the final results in an educational setting. Although simple, this example represents a large class of collaborative applications developed as a part of multi-domain sciences and motivates the ongoing research activities in the Grid community.

The administrative overhead resulting from *many* such individual and sporadic experiments makes it impractical for such transient communities (possibly one-time collaboration) to undergo a formal Grid establishment process. Thus, without a coordinating entity, no single participating individual can be entrusted with the administrative privileges of such a short-lived Grid. Nonetheless, the contributed services and the shared resources must be protected from various hostile elements disguised in such open interactions.

Participants need to formulate and enforce their individual usage and security policies protecting their resources from unwanted or hostile peers. Individuals can participate in such collaborations as long as they have the appropriate access privileges to consume resources controlled by peers. A distributed policy enforcement scheme will provide a robust and scalable solution to the Grid establishment and control problem in transient collaborations.

B. Grid Markets

A Grid market is an important use case being actively researched within the Grid community [13], [14]. A Grid market is a framework in which a Grid resource (computational cycles, data storage, network bandwidth, and specialized services) is treated as a commodity. Individuals or organizations participate in a Grid market by trading their resources with a potential resource consumer. Participating entities negotiate pricing policies and service quality with the ultimate goal of optimizing their respective objective functions. Due to economic implications, Grid markets are inherently competitive (potentially hostile) in nature. Nevertheless, they provide the requisite decentralized brokering infrastructure for bridging the gap between geographically separated resource providers and consumers.

Every participating entity has its objective function, negotiating principles, and usage policy. Thus, Grid markets cannot be regulated and monitored by a single controlling authority. Further, Grid markets have a metamorphic structure. Due to its self-organizing principles, the organizational structure of a Grid market is reflected by its participants, who are in flux. Conventional Grid architectures fail to support such self-organizing communities because they rely on network- and structure-dependent services. Grid markets need a decentralized, self-organizing, self-enforcing, and self-monitoring Grid architecture that enables the independence, security, and robustness desired by participants in order to efficiently trade their resources.

III. Ad Hoc Grids

Extensive research has been conducted on ad hoc networks, an adaptive wireless communication infrastructure between power-constrained devices [15]. However, in the context of ad hoc Grids, we focus on the sporadic and ad hoc nature of the Grid structure, protocols, and control rather than the mobility of devices. Informally, we define an ad hoc Grid as *a distributed computing architecture offering structure-, technology-, and control-independent Grid solutions* that support sporadic and adhoc use modalities.

Structural independence in an ad hoc Grid reflects its ability to self-organize without synchronous coordination between participating entities. Unlike traditional Grid frameworks with well-known Grid entry points, such as a Web page for Grid account requests [16] and a central Grid information index server for service discovery, an ad hoc Grid does not have any formal, well-defined, or agreed-upon entry point.

Instead, peers can join an ad hoc Grid as long as they can discover another member participating in that Grid. In other words, every member of the ad hoc Grid represents an entry point. Several mechanisms for discovering peer entities in the absence of any centralized coordination have been researched by the peer-to-peer community [17]. Ad hoc Grids do not rely on any specific discovery mechanism and can employ multiple solutions simultaneously to improve its efficacy in peer discovery. However, the resilience of ad hoc Grids in terms of avoiding subgroup partitions depends on the discovery solution chosen.

Structural independence in ad hoc Grids provides several benefits lacking in traditional Grid frameworks. It avoids a single point of failure. By offering multiple entry points, the existence of ad hoc Grids is not affected by the unavailability of any single or a group of participants, including the entity that established the ad hoc Grid. It enables the participating peers to establish Grids and collaborations on the fly without depending on any external infrastructure for assistance.

The enthusiasm within the Grid community to provide sophisticated Grid solutions has yielded several Grid technologies [18], [19]. Lack of interoperability between these technologies, however, has resulted in an undesired partition within the Grid user community. Although satisfactory in several scenarios, such lack of interoperability is not acceptable in an ad hoc Grid framework. Ad hoc collaborators may not synchronously agree on the use of a specific Grid technology while establishing a Grid on the fly. Technology independence in an ad hoc Grid reflects its ability to support diverse Grid technologies and protocols.

Control independence in ad hoc Grids signifies its ability to manage its security and usage policies in the absence of a central controller. Due to its structural independence, any peer in an ad hoc Grid cannot rely on external support for crucial services. Thus, the centralized administrative services in traditional Grids that are responsible for membership, access, and usage control on Grid resources are segregated to be hosted on every participating peer. Every entity in an ad hoc Grid is responsible for maintaining and securing its respective resources. Depending on internal policies, participants may allow universal access or restrict access to a few trusted peers.

IV. The Ad Hoc CoG Kit Framework

Ad hoc Grids are not intended to replace any of the existing Grid architectures. At the same time, minor modifications to existing Grid solutions cannot satisfy the requirements of the ad hoc Grid frameworks. Commodity technologies such as the project Jxta [20] and the current modules contained in the Java CoG Kit [21] provide solutions to different aspects of ad hoc Grids. However, a comprehensive and robust infrastructure specifically targeted to solving real problems with ad hoc Grid paradigms is not yet available. To provide such an infrastructure, we introduce a framework that aggregates key technologies, abstractions, interfaces, services, and models [22] to enable real-time ad hoc Grid computing (see Figure 1). The framework also focuses on essential research issues that play an important role in any decentralized, self-organizing, and resource-sharing architecture.

Rather than re-inventing a scalable, flexible, and extensible self-organizing infrastructure, the framework employs the Jxta technology [20] to enable its structure-independent objectives. Jxta is a collection of open peer-to-peer protocols and services that allow any device with

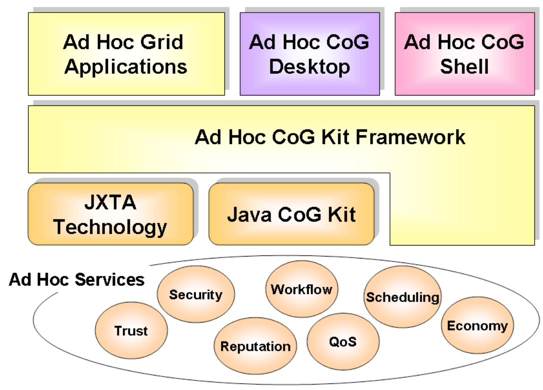

Fig. 1. The proposed *ad hoc CoG Kit* framework reuses key commodity technologies such as the project Jxta and traditional Grid protocols and services. It also contributes several high level services to enable a robust, self-sustaining ad hoc Grid architecture.

a "network heartbeat" to communicate and collaborate with other peers autonomously. It provides a mechanism to create virtual ad hoc collaborations without exposing any of the underlying peer-to-peer protocol complexities. It enables the formation of a self-organizing super-peer-based overlay network on the Internet. Further, it allows a completely decentralized advertisement and discovery of peers and services using distributed hash tables [23].

Using Jxta, our framework creates an overlay network that we referred to as a *ad hoc community Grid*. Per definition, all peers are members of the publicly available ad hoc community Grid. The ad hoc community Grid serves as the pervasive self-organizing infrastructure within which peers can establish their ad hoc collaborations. On joining the ad hoc community Grid, peers can create virtual organizations (VOs) or join existing VOs created by other peers.

Peers can share services, exchange data, and interactively communicate with other peers within the same VO. Thus, by using Jxta, the framework concentrates on problems related to the integration of ad hoc paradigms into the Grid domain, rather than on core peer-to-peer deployment issues.

Experience gained from application requirements over the last decade by our team has resulted in the creation of a suite of pattern-based Grid abstractions [24] that shield from the technical and semantic complexities of various Grid technologies [18], [19]. These abstractions are part of the Java CoG Kit [21]. Applications using these abstractions can interface with different Grid technologies without much effort. Key to enabling the ad hoc Grid framework is the Java CoG Kit abstraction layer, which is reused to enable technology-specific Grid interactions.

The mere combination of the Jxta technology and the Java CoG Kit does not necessarily result in a secure, reliable, and self-sustaining ad hoc Grid framework. Although an ad hoc Grid must support structure-, technology-, and control-independence, it is more important that such a Grid deliver practical Grid solutions in a dynamic environment. Some of the most elementary assumptions in traditional Grid environments regarding trust, reputation, and stability do not hold true in ad hoc frameworks. Hence, several important concepts of Grid computing must be revisited. Although a detailed discussion of all the components in the ad hoc Grid framework is beyond the scope of this paper, for completeness we briefly outline some important services that collectively provide a robust Grid solution in an ad hoc setting.

- Security: Being technology-independent, the ad hoc Grid framework must support various security solutions for authorization and authentication associated with different Grid technologies. It must also protect Grid services from malicious peers, and protect data from malicious services. Another aspect of security in a competitive environment is to verify the quality and validate the quantity of remote services offered [25].
- Trust and Reputation: In the absence of a globally trusted authority, participating peers must explicitly establish and maintain a trust relationship among themselves. The trust and reputation service builds a distributed confidence network that promotes fair play in a potentially hostile environment [26].
  It provides a measure of "goodness" of the participating peer, thereby motivating peers to honor their commitments and implement their policies to improve their respective reputations.
- Quality of Service (QoS): The only realistic assumption in an ad hoc Grid is existence of an unreliable "best effort" environment. No predictions can be made regarding the connectivity and service capability of the participating peers. Traditional Grid solutions cannot be offered in such sporadic environments. For example, it may be impractical for resource consumers to repeat their computations with the same resources because several resource providers decided to disconnect their resources from the ad hoc Grid. To offer satisfactory Grid solutions in an unreliable environment, resource providers must offer explicit QoS assurances regarding availability, stability, and capability [27]. The ad hoc Grid framework includes QoS services that provide a mechanism for resource reservation, quality and pricing negotiation, QoS-enabled service invocation, and QoS agreement enforcement. To make the QoS services more reliable the information returned as part of the service level agreement may be itself weighted and introduce a Quality of Information [12].
- Economy: One of the biggest concern in open infrastructures is the "tragedy of the commons," over consumption of a few popular goods [28]. The economy service [29] implements key economic engineering principles in the ad hoc Grid architecture preventing this dilemma. Assigning physical costs to service usage prevents excessive use of important services. At the same time, monetary profits from service provision encourages service providers to improve their quality, thereby resulting in an improved and more predictable Grid environment.
- Resource Scheduling: One of the most crucial service in a Grid environment is the scheduling service. It is responsible for selecting a Grid task and matching it with the most appropriate Grid resource, optimizing some objective function. The scheduling service is responsible for optimizing the multivariate objectives of the peer considering the unpredictable nature of resource availability. For example, a peer can set a scheduling policy for the resource scheduler asking it to select appropriate Grid services such that it has high reputation, good QoS provision, low cost of invocation, and can complete the task within a specified period.
- Workflow: Key to the success of a Grid framework is its ability to orchestrate and translate complex task ordering and dependencies [30]–[32]. The workflow service enables an advanced execution system that allows the formulating of complex task ordering in an unstable and dynamic environment. Execution flows include directed acyclic graph-like control and data dependencies. To adapt itself to the unreliable ad hoc environment, the workflow service also implements fault-tolerant checkpointable workflows.

Although not every component of the ad hoc Grid services are implemented at this time they provide an initial step for making ad hoc Grids a reality. Rather than focusing on a single aspect of Grid or peer-to-peer computing, it aims at providing a comprehensive infrastructure combining the advantages of both paradigms. We term the set of components that build the ad hoc Grid

V. Summary

Existing Grid architectures can be categorized into national Grids, project Grids, enterprise Grids and volunteer Grids. Although these architectures support various applications with diverse scope and requirements, they fail to support sporadic collaborations in the absence of a central regulating authority. Motivated by the need to support such applications, we introduce the ad hoc Grid architecture.

Ad hoc Grids offer a structure-, technology-, and control-independent Grid solution. Structural-independence reflects the ability to self-organize among its participant peers. Technology independence reflects the ability to support multiple Grid protocols and technologies. Control independence embodies the ability to support administrative functionality without any central coordination. Applications changing members, policies, and requirements are well suited for ad hoc Grids.

We also introduce the ad hoc CoG framework to address some of the critical research issues associated with self-organizing, adaptive, and unreliable distributed frameworks. The framework combines essential commodity technologies such as project Jxta and the Java CoG

Kit. It also provides several utility Grid services to enable self-sustaining ad hoc collaborations. Some of the most important services of the ad hoc CoG framework will be autonomic security service, trust and reputation service, QoS service, economic engineering service, adaptive scheduling service, and workflow service.

ACKNOWLEDGMENTS

This work was supported by the Mathematical, Information, and Computational Science Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract W-31-109-Eng-38. DARPA, DOE, and NSF support Globus Project research and development. The Java CoG Kit Project is supported by DOE MICS, and NSF Alliance.

REFERENCES

[1] G. von Laszewski and P. Wagstrom, *Tools and Environments for Parallel and Distributed Computing*, ser. Series on Parallel and Distributed Computing. Wiley, 2004, ch. Gestalt of the Grid, pp. 149–187. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--gestalt.pdf

[2] "GridCafe: The place for everybody to learn about the Grid," Web Page. http://gridcafe.web.cern.ch/gridcafe/

[3] "German D-Grid Initiative Announcement," Web Page. http://www.gridtoday.com/04/0315/102836.html

[4] "IBM and China's Ministry of Education Launch 'China Grid'," IBM Press Release, October 2003.

[5] "GriPhyN - Grid Physics Network," Web page. http://www.griphyn.org/index.php

[6] M. Russell, G. Allen, I. Foster, E. Seidel, J. Novotny, J. Shalf, G. von Laszewski, and G. Daues, "The Astrophysics Simulation Collaboratory: A Science Portal Enabling Community Software Development," *Journal on Cluster Computing*, vol. 5, no. 3, pp. 297–304, July 2002. http://www.mcs.anl.gov/~gregor/papers/astro-jcc.pdf

[7] "Particle Physics Data Grid," Web Page, 2001. http://www.ppdg.net/

[8] A. Chien, B. Calder, S. Elbert, and K. Bhatia, " Entropia: Architecture and Performance of an Enterprise Desktop Grid System," *Journal of Parallel and Distributed Computing*, vol. Special Issue on Grids and Peer to Peer Systems, 2003.

[9] E. Korpela, D. Werthimer, D. Anderson, J. Cobb, and M. Leboisky, "SETI@home-massively distributed computing for SETI," *Computing in Science & Engineering*, vol. 3, no. 1, pp. 78–83, January–February 2001.

[10] K. Amin, G. von Laszewski, and A. R. Mikler, "Grid Computing for the Masses: An Overview," in *Grid and Cooperative Computing (GCC2003)*, Shanghai, China, December 2003, pp. 464–473. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--masses-gcc03.pdf

[11] G. von Laszewski, M.-H. Su, J. A. Insley, I. Foster, J. Bresnahan, C. Kesselman, M. Thiebaux, M. L. Rivers, S. Wang, B. Tieman, and I. McNulty, "Real-Time Analysis, Visualization, and Steering of Microtomography Experiments at Photon Sources," in *Ninth SIAM Conference on Parallel Processing for Scientific Computing*, San Antonio, TX, 22-24 Mar. 1999. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--siamCmt99.pdf

[12] G. von Laszewski, J. Gawor, C. J. Peña, and I. Foster, "InfoGram: A Peer-to-Peer Information and Job Submission Service," in *Proceedings of the 11th Symposium on High Performance Distributed Computing*, Edinbrough, U.K., 24-26 July 2002, pp. 333–342. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--infogram.ps

[13] R. Buyya, D. Abramson, and J. Giddy, "An Economy Driven Resource Management Architecture for Global Computational Power Grids," in *The 2000 International Conference on Parallel and Distributed Processing Techniques and Applications*, Las Vegas, USA, 26-29 June 2000.

[14] "A Market for Grid Services," Web Page. http://www.lesc.ic.ac.uk/markets/

[15] "Mobile Ad-hoc Network (MANET) Charter," Web Page. http://www.ietf.org/html.charters/manet-charter.html

[16] "TeraGrid," Web Page, 2001. http://www.teragrid.org/

[17] D. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu, "Peer-to-peer computing," HP Labs, Tech. Rep. HPL-2002-57, 2002.

[18] "The Globus Project," Web Page. http://www.globus.org

[19] D. Thain, T. Tannenbaum, and M. Linvy, *Grid Computing: Making the Global Infrastructure a Reality*. John Wiley, 2003, no. ISBN:0-470-85319-0, ch. Condor and the Grid.

[20] "Project JXTA," Web Page. http://www.jxta.org/

[21] G. von Laszewski, I. Foster, J. Gawor, and P. Lane, "A Java Commodity Grid Kit," *Concurrency and Computation: Practice and Experience*, vol. 13, no. 8-9, pp. 643–662, 2001. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--cog-cpe-final.pdf

[22] G. von Laszewski and K. Amin, *Grid Middleware*. Wiley, 2004, ch. Chapter 5 in Middleware for Commnications, pp. 109–130. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--grid-middleware.pdf

[23] B. Traversat, M. Abdelaziz, and E. Pouyoul, "A Loosely-Consistent DHT Rendezvous Walker," Sun Microsystems, Inc, Tech. Rep., March 2003.

[24] K. Amin, M. Hategan, G. von Laszewski, and N. J. Zaluzec, "Abstracting the Grid," in *Proceedings of the 12th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP 2004)*, A Coruña, Spain, 11-13 Feb. 2004, pp. 250–257. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--abstracting.pdf

[25] P. Golle and I. Mironov, "Uncheatable distributed computations," *Lecture Notes in Computer Science*, vol. 2020, p. 425, 2001.

[26] G. von Laszewski, B. Alunkal, and I. Veljkovic, "Toward Reputale Grids," *(to be published)*, 2004. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski-pdcp-reputation.pdf

[27] R. Al-Ali, K. Amin, G. von Laszewski, M. Hategan, O. Rana, D. Walker, and N. Zaluzec, "QoS Support for High-Performance Scientific Applications," in *Proceedings of the IEEE/ACM 4th International Symposium on Cluster Computing and the Grid (CCGrid 2004)*. Chicago IL, USA: IEEE Computer Society Press, 2004. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--qos-ccgrid04.pdf

[28] A. Oram, Ed., *Peer-To-Peer: Harnessing the Power of Disruptive Technologies*, 1st ed. O'Reilly, 2001.

[29] S. Vazhkudai and G. von Laszewski, "A Greedy Grid - The Grid Economic Engine Directive," in *First International Workshop on Internet Computing and E-Commerce (ICEC'01)*, San Francisco, California, USA, 27 Apr. 2001, p. electronically published. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski-greed.pdf

[30] G. von Laszewski, K. Amin, S. Hampton, and S. Nijsure, "GridAnt – White Paper," Argonne National Laboratory, Tech. Rep., 31 July 2002. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski-gridant.pdf

[31] K. Amin, M. Hategan, G. von Laszewski, N. J. Zaluzec, S. Hampton, and A. Rossi, "GridAnt: A Client-Controllable Grid Workflow System," in *37th Hawai'i International Conference on System Science*, Island of Hawaii, Big Island, 5-8 Jan. 2004. http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--gridant-hics.pdf

[32] M. Hategan, G. von Laszewski, and K. Amin, "Karajan: A grid orchestration framework," Supercomputing 2004, Pitsburgh, 6-12 Nov. 2004, (Refereed Poster). http://www.sc-conference.org/sc2004